Incorporating Risk-Sensitiveness into Feature Selection for Learning to Rank

Daniel Xavier de Sousa
UFMG–DCC, Brazil
danielxs@dcc.ufmg.br

Sérgio Daniel Canuto
UFMG–DCC, Brazil
sergiodaniel@dcc.ufmg.br

Thierson Couto Rosa
UFG–INF, Brazil
thierson@inf.ufg.br

Wellington Santos Martins
UFG–INF, Brazil
wellington@inf.ufg.br

Marcos André Gonçalves
UFMG–DCC, Brazil
mgoncalv@dcc.ufmg.br

ABSTRACT

Learning to Rank (L2R) is currently an essential task in basically all types of information systems given the huge and ever increasing amount of data made available. While many solutions have been proposed to improve L2R functions, relatively little attention has been paid to the task of improving the quality of the feature space. L2R strategies usually rely on dense feature representations, which contain noisy or redundant features, increasing the cost of the learning process, without any benefits. Although feature selection (FS) strategies can be applied to reduce dimensionality and noise, side effects of such procedures have been neglected, such as the risk of getting very poor predictions in a few (but important) queries. In this paper we propose multi-objective FS strategies that optimize both aspects at the same time: ranking performance and risk-sensitive evaluation. For this, we approximate the Pareto-optimal set for multi-objective optimization in a new and original application to L2R. Our contributions include novel FS methods for L2R which optimize multiple, potentially conflicting, criteria. In particular, one of the objectives (risk-sensitive evaluation) has never been optimized in the context of FS for L2R before. Our experimental evaluation shows that our proposed methods select features that are more effective (ranking performance) and low-risk than those selected by other state-of-the-art FS methods.

Keywords

Learning to Rank, Feature Selection, Risk-Sensitiveness

1. INTRODUCTION

Learning to Rank (L2R) has established itself as an important research area in Information Retrieval (IR). To obtain good results, L2R strategies usually rely on dense representations exploiting dozens of features, some of which are expensive to generate. In several scenarios, some of these features may introduce noise, or may be redundant, which increases the cost of the learning process without bringing benefits. As a result, one of the research lines in this area is to reduce costs while trying to improve effectiveness.

Feature Selection (FS) techniques have already been examined in the L2R scenario [25, 10, 23]. FS approaches aim to reduce the number of features to improve processing time and to increase effectiveness, while reducing noise and redundancy. More specifically, the FS task may have a high impact on processing time in L2R. In addition to the training time, there is also the cost of constructing the features (actually meta-features) as they are generated by several algorithms (e.g., BM25, PageRank), some of which need to be computed at query time.

However, effectiveness (that is, ranking performance) and cost (better summarized by the number of exploited features) are not the only objectives one may want to optimize in a L2R task. In fact, recently the **risk** of getting very poor effectiveness for a few queries with a learned model has gained much attention [18, 6, 15]. As discussed in [5], users tend to remember the few failures of a search engine very well rather than the many successful searches [5]. Using FS in L2R may increase this risk. This is because FS reduces the feature space when considering only effectiveness or cost as objectives. Thus, it is possible that using fewer features the ranking of documents worsens for a few (but important) queries, despite improving for many others.

The goal of *risk-sensitive L2R task* is to enhance the quality of a ranking system, while reducing the risk of poorer performance than a corresponding baseline system for any given topic. Indeed, [21] clearly shows that improvements in ranking performance do not always correlate with risk reduction. This has motivated research considering the risk aspect in L2R models [7, 6]. However, risk-sensitiveness (or robustness[1]) has not been considered in the context of FS for L2R, as far as we know.

In this paper we propose novel FS methods for L2R, using multi-objective criteria which aim to (sometimes, simultaneously): (i) maximize the ranking performance; (ii) minimize the risk for most queries; and (iii) reduce the feature space dimensionality. Notice that some of these requirements may be conflicting: The removal of a particular feature may reduce the learning cost or increase the overall effectiveness

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CIKM'16 , October 24-28, 2016, Indianapolis, IN, USA
© 2016 ACM. ISBN 978-1-4503-4073-1/16/10...$15.00
DOI: http://dx.doi.org/10.1145/2983323.2983792

[1] From now on, these two terms will be used as synonyms.

of the learned model. However, the resulting ranking model may not perform well for some specific queries, increasing the risk. For that reason, the main goal of this work is to contribute with FS task for L2R, considering risk-sensitive evaluation mainly, a topic which has not been studied before in this context.

Contrary to existing FS approaches for L2R aimed to drastically reduce the number of features to control noise and redundancy, we here focus on how to produce better models even when performing FS. Indeed, our proposal uses different multi-objective combinations, for instance, using ranking and risk measures as objective criteria. Therefore, instead of learning a model with the smallest set of features, we look for a model with a reduced set of features that guarantees both good ranking and a risk-sensitive performance.

Our proposal uses a multi-objective criteria approach based on Pareto frontier optimization. There are several general-purpose multi-objective optimization methods that can be used in this case. We have chosen the Strength Pareto Evolutionary Algorithm (SPEA2) [9], which has already been successfully used in related problems [8, 4]. Moreover, one of the key points in this work is the improvement of the fitness score in order to take into account the multi-objective criteria and the distance between distinct subsets of features.

Our solution evaluates a huge search-space by looking for the best individuals (subsets of features) that can optimize a pair of objectives. Unlike other FS methods, our method is not attached to a specific L2R algorithm, and does not use specific heuristics which may be tailored to some datasets. Furthermore, most of previous work on FS for L2R has shown results only for small datasets, while in this work we consider datasets of varying sizes, including the larger MSLR-WEB10K and Yahoo! datasets.

We claim that by using a multi-objective feature selection considering risk as one of its objectives, it is possible to perform a robust FS that minimizes the degradation of a L2R method regarding some queries. However, as our multi-objective FS strategy can be used along with different combinations of objectives, we also perform an extensive analysis showing how different combinations impact on the final goal of FS.

The experimental results show that our multi-objective FS proposal outperforms all FS methods evaluated. Considering the objective criteria, we are able to: (i) obtain large reductions in dimensionality without significant losses in effectiveness; (ii) reduce the dimensionality by *very large* margins (over 84%) if some losses in effectiveness are acceptable due to the cost of the L2R process; (iii) reduce risk without degrading effectiveness, while still being able to reduce dimensionality.

In summary, the main contributions of this paper are:

- FS methods for L2R which optimizes multiple, in some cases conflicting, criteria. One such objective (risk-sensitive evaluation) has never been optimized in the context of a FS method for L2R.
- A method to select the Pareto-optimal set for the task of FS in L2R, using a strict paired test comparison and a biased L2R algorithm.
- A thorough comparative evaluation of our methods, considering several baselines, objective criteria and different datasets.

Below, we present related work in Section 2. In Section 3 we describe our proposal, and provide an analysis of our experiments in Section 4. Finally, Section 5 summarizes our conclusions, suggesting future works.

2. BACKGROUND AND RELATED WORK

2.1 Feature Selection in L2R

In this section, we review a few FS strategies for L2R in order to contextualize our work with regards to the literature. For more information on this theme we suggest [17].

There are several works exploiting FS for L2R. They can be divided into three main strategies: Embedded, Filter, and Wrapper. Embedded strategies make the selection of features while trying to minimize the training error during the learning of the model. The objective function searches for the minimal best subset of features using a specific L2R error function. For instance, in [12] an Embedded strategy called FSMRank is proposed, which attempts to minimize the ranking errors while performing FS using a combination of importance and similarity measures. However, as asserted in [17], these Embedded solutions are designed for specific L2R algorithms, making them hard to adapt to other L2R alternative methods. This is an important limitation, as the area evolves with better L2R solutions.

Both Filter and Wrapper strategies perform the FS as a step previous to the learning of the model. The selected features are then used to learn a model using some L2R method. In [25], the authors present several Filter methods that select the most relevant and at the same time diverse features, applying diversification techniques [2], for example, Maximal Margin Relevance (MMR), and a variation named Minimum Redundancy Maximum Relevance (mRMR)[13]. More recently, work in [27] has further evaluated the mRMR concept, considering a non-linear feature selection method for L2R. They select a subset of k features (a predefined parameter) such that relevance and dissimilarity among the features are optimized. However, in these works, the importance of a feature is computed one at a time. As [10] shows, the worth of a feature depends on the set of other features with which it interacts. Thus, unlike these works, we deal with sets of features, and not individual features.

Wrapper strategies perform the selection of features based on the effectiveness of a "generic" L2R method (known as a black-box method) which is optimized by the learning procedure with the selected features. The black box method computes the worthiness of a subset of features prior to the learning of the model. For instance, the authors in [10] propose a wrapper strategy based on Genetic Algorithms (GA). They perform the evolutionary process by eliminating "weak" features over the generations in order to reduce the dimensionality. As a result, they can achieve a reduced number of features with small losses in effectiveness. Furthermore, the work shows the importance of considering the interaction of features in order to apply the FS. However, the final number of features has to be set as a parameter, which is difficult to determine. Besides this, the main solution optimizes only one objective – relative feature importance.

Wrapper solutions are agnostic to a particular L2R method or dataset, making them adaptable solutions. The method proposed in this article corresponds to a wrapper solution. However, instead of using a state-of-the-art L2R method during FS, as is the usual case, we opted for a weak learner.

This approach led to a substantial improvement in the quality of the selected features, as will be discussed in Section 3.2.3.

Our wrapper strategy also differs from previous ones in that it exploits a multi-objective Pareto-efficient method. Thus, in order to reach our goals we adapted the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [9], which besides using a wrapper strategy, is also a general-purpose multi-objective criteria method. In [8], authors use SPEA2 with two competing criteria: minimizing the number of features while maximizing effectiveness for the task of determining the quality of collaborative content on the Web.

2.2 Risk and Risk-Sensitive Evaluation

Suppose that we are given a set of training queries Q_T, and two ranking models: a baseline B and a proposed model M. The *risk* of model M corresponds to the average gain in effectiveness of the baseline B against M in all queries in Q_T. This definition of risk was formally stated in [18] through the F_{RISK} function defined in Eq. 1.

$$F_{RISK}(Q_T, M) = \frac{1}{|Q_T|} \sum_{Q \in Q_T} max[0, B(Q) - M(Q)] \quad (1)$$

where $B(Q)$ and $M(Q)$ denote the effectiveness values of the baseline and of the new model, respectively, for a given query Q. The effectiveness can be measured by any commonly-used IR evaluation measure (e.g. MAP, MRR, NDCG@k [24]). Note that as the value of function F_{RISK} decreases, it improves the chance of having a robust model. In this paper we adopt the function F_{RISK} as the definition of the risk of a model M with regard to a baseline B.

Risk is an important concept in ranking systems. In [5], the authors argue that the few failures a search engine makes are more noticed by the users than the many successful searches. The same authors also performed an ample study on user experience in recommender systems, finding out how negative high-variance is for the users. Consequently, the minimization of risk has attracted the attention of researchers as an important additional objective for a ranking solution [18, 7, 6]. Furthermore, according to [18] and [15], *robustness* is the ability of a ranking solution to minimize the risk.

Contrary to risk, the *reward* [18] of a proposed method M in relation to a baseline model B, is defined as the average gain in effectiveness of the model M against the baseline B in all queries in Q_T. In [18] reward is formally stated by the function presented in Eq. 2

$$F_{REWARD}(Q_T, M) = \frac{1}{|Q_T|} \sum_{Q \in Q_T} max[0, M(Q) - B(Q)] \quad (2)$$

Reward and risk can be combined in different ways to evaluate how much a method M is sensitive to risk. Sensitiveness to risk is a quality of ranking systems that has recently attracted the attention of L2R researchers. The term *risk-sensitive task* [15] was coined in The TREC 2013 Web track as the trade-off a system can achieve between effectiveness (overall gains across queries) and robustness, both regarding a baseline [18, 6]. In other words, a method is risk-sensitive if it can improve most queries and does not decrease the ranking performance of others with respect to a baseline (from now on referred to as *risk-baseline*). Thus, the risk-sensitive task corresponds to a multi-objective optimization solution for the ranking problem which aims to maximize effectiveness and minimize the risk[2].

In [18], a measure to evaluate sensitiveness to risk is defined by means of the function U_{RISK} which aggregates functions F_{RISK} and F_{REWARD} in a single *tradeoff function.* U_{RISK} is the objective function that the proposal in [18] aims to maximize. Function U_{RISK} is defined as:

$$U_{RISK}(Q_T, M) = F_{REWARD}(Q_T, M) - (1 + \alpha)F_{RISK}(Q_T, M) \quad (3)$$

The parameter α is the weight given to the risk (F_{RISK}). Different values of α can significantly impact the risk-sensitive evaluation of the method.

The work described in [6] extends the work in [18] by proposing a generalization of the U_{RISK} function which is referred to as T_{RISK}.

$$T_{RISK}(Q_T, M) = \frac{U_{RISK}(Q_T, M)}{SE(U_{RISK}(Q_T, M))} \quad (4)$$

where SE is the estimation of the U_{RISK} standard error.

The proposed function, T_{RISK}, uses inferential hypothesis testing for evaluating risk-sensitive task. The inferential techniques proposed in the paper enable us to: a) decide whether an observed level of risk for an IR system is statistically significant, and b) determine the queries that individually lead to a significant level of risk.

On the other hand, the authors of [7] study how the ranking method used as risk-baseline can affect the risk-sensitive evaluation. They show that the choice of an appropriate risk-baseline is of great importance in ensuring an unbiased risk-sensitive measurement of the performance of individual systems. In particular, the higher the correlation between any given system s and the risk-baseline system across queries, the higher the measured risk-sensitive scores of s on average. This implies a bias in the estimation of the risks. The paper suggests some unbiased baselines, such as mean or maximum ranking performance over several ranking methods.

All the aforementioned works aim to enhance the risk-sensitive task without considering FS. In this article we propose FS strategies that make use of a multi-objective Pareto efficient method to optimize the trade-off between effectiveness and risk while obtaining a reduction in dimensionality. In addition to trying to reduce the feature space, our approach differs from previous work [18, 7] in that it optimizes both ranking performance and risk minimization without the need for a predefined alpha (α) parameter.

3. FEATURE SELECTION PROPOSAL

3.1 Motivation

In this work, we propose a FS method which searches for a subset of the features that satisfy three properties: (i) being small, (ii) improving the ranking performance, and (iii) reducing the risk. In many ways, these three objectives conflict and we do not expect to find a solution that optimizes all criteria. For instance, as shown in Section 4, trying to minimize the number of features while optimizing the ranking performance may generate very specialized solutions, with few features fitting a group of queries, increasing the risk of getting poor effectiveness for some other queries. Thus,

[2] Minimizing the risk is equivalent to maximizing robustness.

instead of trying to optimize the three objectives (minimum number of features, effectiveness and risk), we try to answer the following research question: *is there a subset of features capable of maximizing effectiveness and minimizing risk (i.e. optimizing risk-sensitive evaluation) considering different datasets and L2R methods?*

To tackle this research question, we exploit an evolutionary process which attempt to optimize: (*i*) risk and (*ii*) effectiveness, by varying the set of features to be used in the L2R model. Although the number of features is not a direct objective, it tends to be reduced while improving objectives *i* and *ii* due to the elimination of noisy and redundant features in the original feature space. As we shall see in Section 4, this process allows us to obtain a good feature reduction, without harming the risk-sensitive task. In Section 3.2 we present our evolutionary multi-objective proposal to FS which makes use of the SPEA2 [9] algorithm to select a set of features which is able to optimize both effectiveness maximization and risk minimization. We decided to use this general multi-objective method in our proposal because of its successful results in [8, 4].

3.2 Evolutionary Multi-Objective FS

SPEA2 is based on Genetic Algorithm [28] and thus uses an evolutionary approach to explore the solution space for a multi-objective problem. In this process, each solution (also referred to as an *individual*) receives a *fitness value* that scores its worth based on its likelihood of surviving in the next generation. Once the fitness values have been computed for each individual in one generation, the best individuals are selected to take part in the breeding of the next generation. These selected individuals are kept in an archive A_g during generation g. Thus, along the process, the archives work as buckets to keep the best individuals over the generations. On the other hand, the unfit individuals are eliminated during this evolutionary process. After many generations, surviving individuals (or its descendants) tend to be better than the eliminated ones, according to the fitness criteria.

Algorithm 1 describes the original SPEA2. The algorithm takes as input the size n of the population, the size a of the archive, and the number ng of generations. A population, $P_g = \{i_0, ..., i_n\}$, is the set of individuals in a generation g. In our case, each individual corresponds to a binary array (aka, a chromosome). A position in the array is defined as a *gene*. It is 0 when the feature is absent, and 1 otherwise. The algorithm first creates an empty archive A_1 and a population P_1 with n individuals in Lines 1 and 2, respectively.

Once all individuals have been created, the fitness score for each one is computed (Line 3 and 22). When assigning scores to features, SPEA2 must consider the optimization of multiple objectives. Thus, the algorithm uses the *dominance relationship* among individuals to provide the fitness values. Let x and y be two potentially conflicting objectives. Let also i and j be two different individuals. Individual i *dominates* j (denoted as $i \succ j$), if and only if, $(x_i > x_j \wedge y_i \geq y_j) \vee (x_i \geq x_j \wedge y_i > y_j)$. In other words, i dominates j, if i is better than j in one objective, and i is not worse than j in the other one. An individual i is in the Pareto frontier, when there is no other individual j that dominates i. In this case, i is said to be a *nondominated* individual. The *strength* $S(i)$ of an individual i is defined as the number of individuals who are dominated by i, as described in Eq.5.

$$S(i) = \mid \{ j \mid j \in P_g \cup A_g \wedge i \succ j \} \mid \tag{5}$$

where $\mid . \mid$ is the cardinality of a set. Finally, the fitness score of i is computed by Eq. 6.

$$fitness(i) = R(i) + D(i) \tag{6}$$

where,

$$R(i) = \sum_{j \in (P_g \cup A_g) \wedge j \succ i} S(j) \tag{7}$$

$R(i)$ sums the strength of the individuals who dominate i. Note that $R(i) > R(j)$ means that the individual i is worse than individual j, as the individuals that dominate i are stronger than those who dominate j. Thus, the value of $fitness(i)$ is optimized by minimizing $R(i)$. When $R(i) = 0$, no individual dominates i meaning that all individuals with $R(i) = 0$ are the best solutions, i.e., they belong to the Pareto frontier.

Term $D(i)$[3] in Eq. 6 is referred to as *density estimate.* It is used to break ties and is calculated according to Eq. 8:

$$D(i) = 1/(\sigma_i^k + 2) \tag{8}$$

The value 2 is used to ensure that $D(i)$ is less than 1 and to keep the denominator greater than zero [9]. Also, σ_i^k is the distance between individual i and the k^{th} nearest individual in the binary array using the K-nearest neighbor algorithm [22] with the Euclidean Distance. The parameter k is defined as $\sqrt{\mid A_g \mid + \mid P_g \mid}$.

After computing the fitness for each individual, the algorithm defines the D_g and N_g sets, putting inside D_g the individuals which are dominated by other individuals (Line 6), and in N_g all *nondominated* individuals (Line7).

Lines 8-13 of Algorithm 1 define the elitism process, saving in the archive (A_g) all the *nondominanted* individuals of the population. If the archive is full (Line 9), the algorithm removes the individual which is most similar[4] to all other individuals in the archive. This removal is repeated until the size of the archive becomes equal to the the limit a. This approach increases the diversity over genotype. If the archive is not full (Line 13) the algorithm fills the archive with the best individuals in D_g (i.e. individuals that despite being dominated have small fitness).

After A_g is full, the algorithm initializes the next generation archive (A_{g+1}) with A_g (Line 15). Next, it creates a new population P_{g+1}, performing crossover and mutation on individuals of the current archive (A_g). Crossover is performed by using the Tournament Selection method [28] (Line 17), which selects the individuals with highest fitness values, among a few set of individuals chosen at random from A_g. Using the *Two Point Crossover* [28] method, the crossover

[3] Note that $D(i)$ is assigned to promote a large variety of solutions, as it decreases when i is farther from a dense region. In this sense, a higher priority is given to the more distinct individual, stopping the search process from being trapped in a local optimal solution. In addition, an individual with tied $R(i)$ values but in a sparse region will have more chance of surviving to the next generation. This step helps to avoid overfitting, as the algorithms tend to diversity.

[4] We use Euclidean Distance as the similarity measure among individuals.

Algorithm 1 The Original SPEA2 Algorithm.

```
Require: Population size $n$
Require: Size $a$ of Archive ($A_g$)
Require: Number of generation $ng$
Ensure: $A_g$ close to Pareto frontier
    Let $P_g$ = pop. of individuals $\{i_0, ..., i_n\}$ of generation $g$
    Let $A_g$ = the best individuals of all generations until $g$
    Let $D_g$ = dominated individuals of $P_g$ and $A_g$
    Let $N_g$ = non-dominated individuals of $P_g$ and $A_g$
 1: $A_1 \leftarrow \emptyset$
 2: Initialize $P_1$ with random individuals
 3: Compute fitness(i), $i \in P_1$
 4: for $g = 1$ to $ng$ do
 5:     with $i \in P_g \cup A_g$ do:
 6:         Assign $i$ to $D_g$ if fitness(i) $\geq 1$
 7:         Assign $i$ to $N_g$ if fitness(i) $< 1$
 8:     Add $N_g$ to $A_g$
 9:     if $\mid A_g \mid > a$ then
10:         truncate($A_g$)
11:     else if $A_g < a$ then
12:         $k = a - \mid A_g \mid$
13:         Fill $A_g$ with the $k$ best individuals in $D_g$
14:     $P_{g+1} \leftarrow \emptyset$
15:     $A_{g+1} \leftarrow A_g$
16:     while $\mid P_{g+1} \mid -1 < n$ do
17:         Select two individuals $i_x$ and $i_y$ from $A_g$.
18:         $(new\_i_x, new\_i_y) = crossover(i_x, i_y)$
19:         Add $new\_i_x$ and $new\_i_y$ to $P_{g+1}$
20:     fol all $i \in P_{g+1}$
21:         $random\_mutate(i)$
22:     Compute fitness(i), $i \in P_{g+1} \cup A_{g+1}$
```

is performed (Line 18) exchanging a random continuous sequence of genes on two selected individuals.

In Lines 21 and 22, the algorithm applies a random selection for mutation to each individual. The $random\_mutate(i)$ method flips a coin to perform the mutation for an individual i. In positive case, it flips a coin again for each gene in the chromosome corresponding to i, following a Binomial Distribution with a previously defined parameter. We selected this mutation method in order to set a low probability for a mutation process, so that few individuals are mutated. However, when the mutation is performed, it produces a large modification in the chromosome.

After Algorithm 1 is completely executed, it ensures that a set of individuals are in or close to the Pareto frontier, which is a subset of the last archive. In order to select only one individual (as the definite subset of features) from this set, we choose the individual which produces the greatest ranking performance (effectiveness) in the training set.

As far as we know, the use of SPEA2 to optimize the risk-sensitiveness of a L2R model while reducing the number of features has not been reported in the literature before. However, in addition to using SPEA2 we also need to extend it for our needs. Thus, in the following subsections, we describe our main extensions to SPEA2, regarding: (i) the definition of the dominance relationships used, (ii) the use of statistical test to compute dominance relationships, and (iii) the use of a biased learning algorithm as a black-box L2R method.

3.2.1 Dominance relationships

In this paper, we compute fitness using Eq. 6, however, we use different definitions of the dominance relationship ($\succ$) according to the objectives we want to optimize.

As most of the objective pairs we intend to optimize involve effectiveness or risk, we now discuss how we compare two individuals i and j according to these criteria. To compare i and j regarding effectiveness, we use the values of an IR measure directly such as MAP, MRR or NDCG [24]. The measure is obtained from the two models derived from a black-box L2R method using the features present in individuals i and j, respectively. We refer to the effectiveness value of a model learned with features of an individual i as $\mathit{eff}(i)$.

To compare i and j regarding risk, we compute the values of function F_{RISK} (Eq.1) for the two models obtained by using the black-box L2R with features of i and j, respectively. We denote the F_{RISK} computed for the learned model corresponding to an individual i as $F_{RISK}(i)$.

The main pair of objectives we want to optimize is effectiveness and risk. In this case, we use Definition 1 to determine if an individual i dominates individual j (i.e., $i \succ j$).

Definition 1. $i \overset{E\text{-}R}{\succ} j$ *if and only if* $(F_{RISK}(i) < F_{RISK}(j) \wedge \mathit{eff}(i) \geq \mathit{eff}(j)) \vee (F_{RISK}(i) \leq F_{RISK}(j) \wedge \mathit{eff}(i) > \mathit{eff}(j))$.

It is worth noting that by using $\mathit{eff}(i)$ and $F_{RISK}(i)$ as independent objectives, we are improving the computation of risk-sensitive evaluation in comparison to those computed by U_{RISK} and F_{RISK}, Eq. 3 and Eq. 4 respectively. This is because we have no parameter α to be adjusted.

We also optimized other objectives. The second pair we analyzed is made up of the risk-sensitive evaluation (T_{RISK}, Eq. 4) and the number of features. Thus, the dominance relationship, in this case, given two individuals i and j, is defined according to Definition 2:

Definition 2. $i \overset{F\text{-}T}{\succ} j$ *if an only if* $(nFeat(i) < nFeat(j) \wedge T_{RISK}(i) \geq T_{RISK}(j)) \vee (nFeat(i) \leq nFeat(j) \wedge T_{RISK}(i) > T_{RISK}(j))$

where $nFeat(i)$ corresponds to the number of features of individual i. We also defined a dominance relationship based only on the risk-sensitive evaluation, i.e., T_{RISK} (Eq. 4), according to Definition 3

Definition 3. $i \overset{T}{\succ} j$ *if an only if* $T_{RISK}(i) > T_{RISK}(j)$

In the case of only one criterion, there is no Pareto frontier, and the SPEA2 algorithm becomes similar to a classic genetic algorithm.

In order to further evaluate our multi-objective approach, we also defined the dominance relationship based on ranking effectiveness (Definition 4) and the dominance relationship based on both ranking effectiveness and number of features (Definition 5). We do not consider the number of features as a single objective, as this has a trivial solution: a single feature.

Definition 4. $i \overset{E}{\succ} j$ *if an only if* $\mathit{eff}(i) > \mathit{eff}(j)$

Definition 5. $i \overset{E\text{-}F}{\succ} j$ *if an only if* $(nFeat(i) < nFeat(j) \wedge \mathit{eff}(i) \geq \mathit{eff}(j)) \vee (nFeat(i) \leq nFeat(j) \wedge \mathit{eff}(i) > \mathit{eff}(j))$.

For each objective O (except number of features) we used a statistical significance test when comparing two individuals according to objective O. In Section 3.2.2 we discuss the importance of theses statistical tests.

3.2.2 *Using of paired tests in the dominance relationships*

One important issue when using Pareto frontier is to select the final individual to learn the definite model. This is because the Pareto frontier obtained is usually large, especially when the two objectives are opposed [16]. To explain why the Pareto set increases, let us consider an example of two non-dominated individuals, i and j. Suppose that regarding objective x, i is a little greater than j, using a scalar value of some measure. Otherwise, taking into account another objective y and also using a scalar value, j can be substantially greater than i regarding y. In this case, there is no dominance relationship between i and j, and as a result, both individuals are kept on Pareto frontier, increasing its size. However, note that there is a small difference between both individuals for objective x.

In this work we dealt with this issue by considering that the models learned for individuals i and j may be used to generate rankings for each query of the training set. Thus we can compare both models per query (with regard to the objective x) and use the training set as a sample for the evaluation. This allows us to perform a paired statistical test (and consequently comparing individuals i and j confidently). Using our aforementioned example, the individual i will probably not be statistically different from j over objective x, thus defining i dominated by j, and consequently, keeping only j as a non-dominated individual.

As our experiments show (see Section 4.2.2), in the cases of datasets with many queries, the test of significance improves comparison of individuals considerably, leading to a strict dominance evaluation. As a consequence, the Pareto frontiers are much smaller than those produced by the conventional method. Furthermore, only high-quality individuals remain in the final Pareto set, improving the selection of the final individual.

3.2.3 *Using a biased algorithm as black-box*

Our wrapper-based FS approach uses SPEA2 to optimize a pair of objectives and a black-box L2R method to compute effectiveness. However, traditional wrapper strategies exploit the same L2R method both as a black-box method during FS processing, and in the final ranking solution where the FS is done. Usually, a state-of-the-art L2R method is chosen. Nevertheless, a high-quality learner (i.e. with both low variance and low bias) is usually able to attenuate the presence of "bad" (i.e., noisy, redundant) features, predicting similar accuracy for different individuals. Measuring an individual's fitness is crucial in our proposal, as it influences the exploration of the search space considerably. Thus, it is more important that the L2R method used inside SPEA2 is able to better qualify a feature set than build a highly effective model. Consequently, for our goals, it is interesting that the L2R method used during FS does not attenuate the effect of bad individuals so that they can be filtered out more effectively from the Pareto frontier during the generation process.

Based on this assumption, during SPEA2 execution we apply a simple Linear Regression as the black-box L2R method to build the model for each individual. Besides improving the comparison between individuals, the Linear Regression method is much faster than most state-of-the-art L2R methods. In Section 4.2.1 we show the great differences between using a biased learner and a state-of-the-art as black-box L2R method in a experiment. Nevertheless, after the final selection of features is done, we apply a more effective method, such as Random Forests[1], MART[3] or LambdaMART [3], all of them well-known, very effective L2R methods.

It is worth observing that this exploitation of more biased learning methods to process a training subspace has been used in different machine learning contexts, such as classifier ensembles. Random Forests [22], for instance, exploits regression trees without pruning, in order to measure the information gain obtained by many training subspaces (aka, bagging). By combining all such biased measurements, it can achieve a model with lower variance [22].

The original SPEA2 does not use some of the improvements proposed in this work, namely: paired difference test, and a biased learning method as a black-box. Moreover, it is the first time that the SPEA2 is evaluated on Feature Selection for L2R. All these improvements and the multi-objective criteria are evaluated in the next section.

4. EXPERIMENTAL EVALUATION

In this section, we describe a set of experiments performed to evaluate our FS proposal. We first describe the datasets, the risk-baselines, the FS baselines methods and the parameter settings. We then provide a description of the objective criteria evaluated. To finalize, we discuss our results and evaluations. Due to space restrictions, we summarize the results of our proposal in this paper.

4.1 Experimental Setup

4.1.1 *Datasets, Algorithms and Procedures*

We conduct our experiments on four well-known benchmark datasets: MSLR-WEB10K (from Microsoft Research[5]), Yahoo! Webscope dataset version 1 and set 2 (from Yahoo! Learning to Rank Challenge[6]), and from LETOR [7] datasets: TD2003 and TD2004. For our evaluation, each dataset was divided into five folds for a 5-fold cross-validation procedure, with three folds for training, one for validation (to search for parameters) and one for testing. The datasets WEB10k, YAHOO, TD2003, and TD2004 have, respectively, 136, 700, 64, and 64 features. With, 10,000, 6,330, 50, and 75 queries, respectively.

In order to ensure the relevance of the results, we assess the statistical significance of our measurements by means of a paired T-test [26] with 95% confidence.

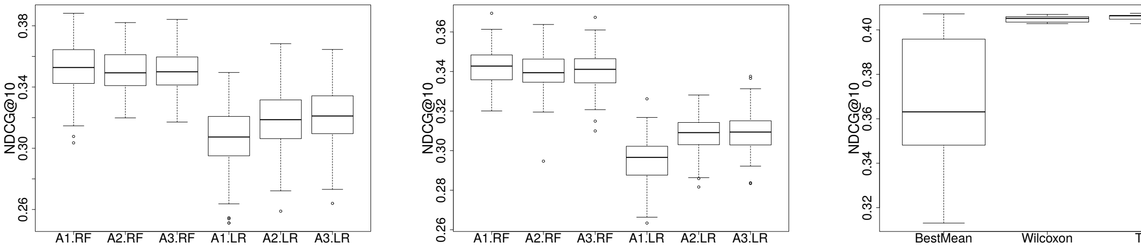
As defined in Section 2.2, the risk-sensitive evaluation is a measure that evaluates the robustness of a ranking solution relative to a defined Information Retrieval (IR) baseline method, or risk-baseline. In order to compute the risk-sensitive evaluation of each individual within SPEA2, we use as the risk-baseline the full set of features combined using a machine learning approach. Because all individuals are evaluated using the same machine learning approach used within SPEA2, this fulfills the requirements defined in [7] to be a valid and unbiased risk-baseline.

To evaluate the risk-sensitive task of our proposal against other methods, as suggested by [7, 18], we use the Mean, Max, and BM25 as baselines. With Mean and Max base-

[5] http://research.microsoft.com/enus/projects/mslr/

[6] http://research.yahoo.com/Academic_Relations

[7] http://research.microsoft.com/enus/um/people/letor/



(a) TD2003

(b) TD2004

(c) Pareto frontiers

Figure 1: *Figures (a) and (b) show the performance (NDCG@10) of SPEA2 using Random Forest(RF) and Linear Regression (LR) for TD2003 and TD2004 datasets. Figure (c) shows the benefits of using the paired tests to improve the ranking performance of the individuals in the Pareto frontier, using the* $\overset{E\text{-}R}{\succ}$ *objective criterion on WEB10k dataset.*

lines we use the value of each feature as a score for a document to be ranked with regard to a given query. Then, for each query we evaluate the effectiveness for each feature (e.g. using NDCG). The Mean baseline corresponds to the mean effectiveness of the features. The Max baseline uses the greatest effectiveness value obtained among the features. The BM25 baseline corresponds to the results of this method applied to the whole document (which is already included as a feature in the datasets). As pointed out in [7], Max and Mean can be seen as unbiased risk-baselines. We note that the risk measures the degradation of a model against some risk-baseline. Therefore it is important that the risk-baseline has a high performance for several queries, otherwise, the weight of the risk would not be evaluated properly. As a result, the Max risk-baseline is an important baseline, and we use it more commonly to compute the risk-sensitive evaluation of the methods.

The chosen machine learning approaches for the SPEA2 internal evaluations follow the discussion in Section 3.2.3, which suggests the need for a highly biased machine learning method. In most datasets, we use linear regression due to its high bias and fast processing time. However, the Yahoo dataset suffers from a high multicollinearity problem[8], which prevents linear regression from producing good models. For this dataset, we use regression trees (without pruning) instead of linear regression, also a biased method. In the other datasets, there are no significant differences between linear regression and regression trees.

With the exception of the alpha parameter of the risk-sensitive measure, which has to be previously established and analyzed in the experiments, some "default" parameterization is necessary for the SPEA2 evolutionary process. In this case, we used some values used in previous work [20, 10, 9]. For instance, [10] indicates that having more individuals per generation is better than having more generations. Hence, we define population size as 75 (as used in [8]) and the number of generations as 30. In addition, as [9] shows, a large archive size suggests a large elitism, thus we use an archive size of 150 (twice the population number). For the mutation and crossover parameters we follow the guidelines [20]: individual mutation probability = 0.2, gene mutation probability[9]= 0.3, and crossover probability =0.8. We apply these same parameters for all SPEA2 executions. In the future, we intend to perform a more thorough analysis of the parameters and their impact on the final solution. In any case, even with this *default* parameterization, we were able to obtain excellent results, as we shall see.

To compute T_{RISK}, we first compute U_{RISK} (Eq. 3) with the following range of α values (considering [18]): 1, 5, 10, 15, 20, 25, 30, and 35. From the best evaluation performance on the validation set, we used $\alpha = 35$ for Yahoo and WEB10k, and $\alpha = 5$ for TD2003 and TD20004.

For a comparative evaluation with other Feature Selection methods, we implemented the works described in [10] (here called BTFS) and in [25] (here called DivFS). They correspond to a wrapper and a filter strategy, respectively, both described in Section 2.1[10]. To evaluate the BTFS method, we selected the best parameters considering the best ranking performance over the validation set. We used the same elimination rate as in the original work: 0.02, 0.10, 0.25 and 0.50. The authors consider that using 30 features of their datasets, which originally contained 419 and 367 features, is enough to obtain the same prediction as using the full dataset. Following them, we use 30, 50, 10 and 10 features, respectively in WEB10k, YAHOO, TD2003, and TD2004. Concerning the evolutionary process in BTFS, we used the same parameters used in our solution. To evaluate the DivFS method, we implemented the best approaches found in the original paper, namely Modern Portfolio Theory and Maximal Marginal Relevance. In both cases, we used the target number of features considering the same rate used by the authors: 0.20, 0.30, and 0.40. We used the validation set only to evaluate the best parameters, and then applied the best values in the test set.

To evaluate the set of features selected by our method, we applied the well-known, state-of-the-art L2R method: Random Forest[22] and LambdaMART[3]. Random Forest is known to be robust to changes in parameters [3], not being influenced very much by them. Therefore, we evaluated the number of trees $\in \{100, 200, 300\}$ on the validation set and left the remaining parameters with their default values (as in the Scikit-Learn[11]). For the learning rate, the number of

[8] We note that from the 700 features in this dataset, 606 of them (about 4 times the number of features in WEB10k) have a VIF (Variance Inflation Factor) value greater than 10. A VIF greater than 10 is a rule of thumb widely used to indicate a high degree of multicollinearity [19].

[9] The *individual mutation probability* is the probability to perform a mutation in a individual, and the *gene mutation probability* is the probability to change a gene.

[10] We did not evaluate the embedded strategy, because of its scalability problems in large datasets.

[11] http://scikit-learn.org/stable/

leaves, and the number of tress of the LambdaMART algorithm, we chose the best performing parameters in the validation set. We evaluated the following values: *learning rate* $\in \{0.025, 0.05, 0.075, 0.1\}$, *number of leaves* $\in \{10, 50, 100\}$, and *number of trees* $\in \{100, 200, 300, 500, 800\}$. The remaining parameters of LambdaMART follow the RankLib[12].

4.1.2 Evaluation Measures

We evaluated the effectiveness for queries of a dataset performing the average of the NDCG@10[24] over all queries[13].

A risk-sensitive evaluation was performed using the T_{RISK} function, with $\alpha = 1$,5 and 10 as suggested in [18, 6]. In our tables, M-T_{RISK} and B-T_{RISK} stand for T_{Risk} evaluated on Mean and BM25 risk-baselines, respectively.

To report the robustness gains regarding the baselines we use F_{RISK} (Eq. 1) and two additional measures: $Wins$ and "$Losses > 20\%$", following [6, 18]. The measure $Wins$ for a method M counts the number of queries for which M wins against the baseline, ignoring ties. The measure "$Losses > 20\%$" expresses the number of queries for which the relative loss in effectiveness of a method M against the risk-baseline is higher than 20%. It is worth noting, that "$Losses > 20\%$" and "F_{RISK}", have a "less is better" interpretation – these are show in our tables using the symbol ↓.

4.2 Results

In this section we evaluate our proposals by showing the performance of the extensions we made in SPEA2 and the result of using our approach varying the objective criteria (including their combinations) in the considered datasets.

4.2.1 Choice of a biased black-box into the SPEA2

We start by confirming, in Figures 1a and 1b, that using a more biased black-box learning algorithm (Linear Regression) rather than a consistent learner (Random Forest) in the evolutionary process improves the exploration of the search-space, as discussed in Section 3.2.3.[14] In fact, a biased learning algorithm may not attenuate the impact of keeping a noisy feature as input. Figures 1a and 1b show the SPEA2 ranking performance (in NDCG@10) on TD2003 and TD2004 datasets, respectively, using a Random Forest and a Linear Regression as learning algorithms. The box-plot in the figures summarizes the ranking performance (NDCG@10) of individuals in the archives from different iterations of the SPEA2 algorithm. We select archives A1, A2, and A3 to represent, respectively, the first, middle and last archives. In plots 1a and 1b, the first three boxes represent the performance using Random Forest, and the last three, using Linear Regression. Random Forest average results are almost constant, while Linear Regression results show an increasing average curve.

This demonstrates that the evolutionary process can take advantage of a more biased learner algorithm to evaluate individuals, thus improving the search for the best results. We note that there is greater variability in ranking performance when using Linear Regression, especially in Figure 1a. This shows that Linear Regression enables a more sensitive evaluation of the individuals. We also obtained similar results with Regression Trees without pruning.

4.2.2 Using paired tests in the dominance relationships

Each box-plot in the Figure 1c summarizes the ranking performance of the individuals in the Pareto frontier, in the last generation. The left box-plot shows the values NDCG@10 for individuals that win the naive comparison by the mean values of NDCG@10. The other two box-plots correspond to comparisons of the mean NDCG@10, but using a paired test to confirm when an individual is superior.

As Figure 1c reveals, there is a great variation in the values when a paired test is not used (left box-plot) due to the large number of individuals in the last archive. In this case, selecting the best individual in this archive is obviously more risky. In contrast, when a paired test is used we are performing a more rigorous comparison between individuals and we keep only the good ones in the Pareto frontier. In these cases, the chance of selecting an individual which is very distinct from the best ones is very small.

The middle box-plot in Figure 1c corresponds to comparisons using the Wilcoxon Signed-rank test [14], whereas the right box-plot corresponds to the t-Test [26]. Their results in Figure 1c are very similar. However, the Wilcoxon test has been shown to be more robust against (or insensitive to) outliers [14]. Consequently for remainder of our experiments the Wilcoxon test was used.

4.2.3 Multi-objective criteria Results

The rows NDCG@10 in Tables 1, 2, and 3 represent the ranking performance for the FS methods evaluated and the full set of features ("Full") on the experiment datasets. We present the results using Random Forests (RF) and LambdaMART as ranking predictors used with the selected features. For the TD2003 and TD2004 datasets in particular, different methods obtained similar results against full features. Indeed, [11] has already demonstrated that it is hard to obtain statistical significance in both datasets because of their low number of queries. However, our goal with these datasets is to highlight the tendencies of ranking performance and risk-sensitive evaluation.

As we can see, the $\overset{\text{E-R}}{\succ}$ multi-objective criteria is the only one that does not lose in both effectiveness and risk-sensitive evaluation, considering any dataset or ranking strategy when compared to the full-feature baseline. This is because considering both criteria (see Definition 1), the optimization process drives the genetic algorithm to a more strict search space containing solutions that improve ranking and risk-sensitive performance at the same time.

This was not the case with the remaining methods as they could not consistently keep statistically indistinguishable results when compared to the full set of features. For the methods which include the number of features as an objective criterion, the genetic algorithm usually ends up at positions in the search spaces containing solutions that damage the ranking performance. As for the FS baselines, DivFS performed more consistently among different datasets, as it ties with full feature more than the BTFS method.

Figure 2 presents the feature space reduction for all evaluated methods. In particular, $\overset{\text{E-F}}{\succ}$, $\overset{\text{T-F}}{\succ}$, and BTFS, reduced the number of features very effectively (varying from an al-

[12] https://sourceforge.net/p/lemur/wiki/RankLib/

[13] We have tested with other metrics such as MAP and NDCG at other positions and the results are qualitatively the same.

[14] It is worth remarking that after the best individual is selected, we use a state-of-the-art algorithm to produce the final rank.

	WEB10k								*YAHOO*							
	Full	$\overset{\text{E}}{\succ}$	$\overset{\text{E-F}}{\succ}$	$\overset{\text{E-R}}{\succ}$	$\overset{\text{T-F}}{\succ}$	$\overset{\text{T}}{\succ}$	DivFS	BTFS	Full	$\overset{\text{E}}{\succ}$	$\overset{\text{E-F}}{\succ}$	$\overset{\text{E-R}}{\succ}$	$\overset{\text{T-F}}{\succ}$	$\overset{\text{T}}{\succ}$	DivFS	BTFS
NDCG@10	0.424	0.421	0.417	$\underline{\mathbf{0.424}}$	0.419	0.422	0.417	0.418	0.702	0.699	0.695	$\underline{\mathbf{0.703}}$	0.699	0.696	0.698	0.699
$T_{RISK}(5)$	-94.7	-95.9	-97.4	$\underline{\mathbf{-94.6}}$	-96.6	-95.5	-96.6	-96.9	-55.6	-56.5	-57.7	$\underline{\mathbf{-55.5}}$	-56.8	-57.4	-56.3	-56.2
$M\text{-}T_{RISK}(5)$	50.7	48.1	46.1	$\underline{\mathbf{49.9}}$	46.9	48.6	44.4	45.3	34.3	32.5	30.2	$\underline{\mathbf{34.2}}$	31.8	31.1	32.8	32.4
$B\text{-}T_{RISK}(5)$	5.6	4.0	2.3	$\underline{\mathbf{5.4}}$	2.7	3.9	1.8	2.3	-	-	-	-	-	-	-	-
$_{FRISK} \downarrow$	0.162	0.165	0.168	$\underline{\mathbf{0.162}}$	0.166	0.164	0.167	0.168	0.106	0.108	0.112	$\underline{\mathbf{0.106}}$	0.109	0.111	0.111	0.108
Loss>20% ↓	5100	5190	5287	$\underline{5133}$	5248	5185	5277	5279	1485	1550	1588	$\underline{1474}$	1533	1554	1518	1527
Win	1750	1744	1630	$\underline{1752}$	1679	1750	1646	1660	1376	1316	1282	$\underline{1404}$	1350	1291	1374	1329

Table 1: *The NDCG@10 and risk-sensitive evaluation in WEB10k and YAHOO datasets, using the RF. Bold represent results statistically indistinguishable with the Full set of features, and underline results the best values between FS methods.*

most 69% to 85% reduction for all datasets), though with a resulting reduction in effectiveness and risk-sensitive evaluation. In cases in which a less drastic reduction takes place, the effectiveness was usually higher. We noted that, $\overset{\text{E-R}}{\succ}$ can reach a significant reduction without degrading the ranking performance and risk-sensitiveness.

Besides feature reduction and ranking performance evaluations, the risk-sensitive results provide valuable information about the robustness of the ranking methods. Tables 1 and 2 provide the risk-sensitive results regarding different measures and well-known L2R datasets. Among the results of different FS strategies, $\overset{\text{E-R}}{\succ}$ is the only method capable of consistently keeping the risk-sensitive evaluation and ranking performance of the "Full Features" model in all datasets, with a considerable feature reduction (up to 26%) in the feature space. This is strong evidence towards our claim that finding a robust solution by exploiting the $\overset{\text{E-R}}{\succ}$ objective criteria is feasible.

The results in our two largest datasets, shown in Table 1, present a clear picture of the robustness of each FS approaches. The $\overset{\text{E-R}}{\succ}$ objective criteria is more robust than other objective combinations in almost all cases, showing that such a combined objective explores the search space with an emphasis on more "robust" individuals. Following our results, when directing the search space to individuals that generate effective and low risk models, it is possible to explore a restrict space of individuals with more chances of accomplishing both objectives, performance and risk. As there is no public description of the features in YAHOO dataset, Table 1 does not contain the BM25 risk-baseline.

In Table 1, the $\overset{\text{E-R}}{\succ}$ objective has the highest values for T_{RISK} in all tests, and the best results in almost all other risk-sensitive measures regarding YAHOO and WEB10k datasets. $\overset{\text{E-R}}{\succ}$ also got very close results with the $\overset{\text{T}}{\succ}$ objective criterion considering the measure "*Losses* $>$ 20%", which counts the number of queries in relation to the Max risk-baseline. Thus, we can conclude that using only a risk measure as objective may be useful to avoid some of the worst results in specific queries.

The risk-sensitive results of the baselines DivFS and BTFS are among the worst in the WEB10k dataset. As for the YAHOO dataset, this picture changes towards relatively better results for BTFS. In fact, most of these FS baselines are statistically indistinguishable with $\overset{\text{E-F}}{\succ}$. This is to be expected as feature selection approaches usually attempt to obtain the best effectiveness and feature reduction, disregarding risk.

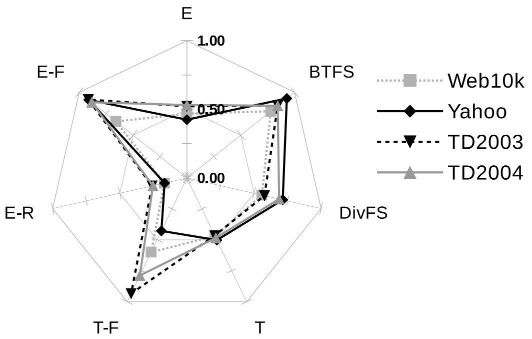


Figure 2: *Description of feature reduction for the methods.*

Table 2 shows the risk-sensitiveness evaluation for TD2003 and TD2004. Since they are very small, it is hard to obtain consistent (i.e., statistically significant) results for them [11]. In fact, many results are statistically indistinguishable with the "Full Features" model. However, $\overset{\text{E-R}}{\succ}$ results are consistently among the best in these datasets, considering all criteria.

We also exploit the LambdaMART algorithm (instead of RF), in Table 3, using WEB10K dataset to evaluate our proposal. These results also confirm the risk-sensitive findings previously discussed: $\overset{\text{E-R}}{\succ}$ is consistently among the best strategies considering different risk-sensitive evaluation.

From our results, by using only ranking performance and/or number of features as an objective, it is not possible to produce robust models. As empirically demonstrated, our proposals build more robust models by explicitly considering the risk-sensitive evaluation as an objective.

On the other hand, we also notice that the best method depends on the main goal. If it is feature reduction, without large performance losses and with low-risk, the best solution is $\overset{\text{T}}{\succ}$ as objective. The reduction was around 47% to 50%, for WEB10K and YAHOO, respectively.

However, if the goal is to obtain a substantial feature space reduction in order to improve the processing cost, but with some harm to effectiveness, the best option is $\overset{\text{E-F}}{\succ}$. In our experiments the reduction in the features space was approximately 66% and 91%, in WEB10K and YAHOO, respectively. Furthermore, the effectiveness was statistically inferior in both datasets, WEB10k and YAHOO.

To conclude, if the main goal is to obtain some reduction in dimensionality in a robust way and without effectiveness losses, the best option is the proposed $\overset{\text{E-R}}{\succ}$ objective with the dimensionality reductions varying from 17 to 26%.

	TD2003								*TD2004*							
	Full	$\overset{E}{\succ}$	$\overset{E\text{-}F}{\succ}$	$\overset{E\text{-}R}{\succ}$	$\overset{T\text{-}F}{\succ}$	$\overset{T}{\succ}$	DivFS	BTFS	Full	$\overset{E}{\succ}$	$\overset{E\text{-}F}{\succ}$	$\overset{E\text{-}R}{\succ}$	$\overset{T\text{-}F}{\succ}$	$\overset{T}{\succ}$	DivFS	BTFS
NDCG@10	0.363	0.326	0.206	$\underline{\mathbf{0.358}}$	0.203	0.324	**0.343**	0.254	0.351	**0.339**	0.282	**0.354**	**0.318**	$\underline{\mathbf{0.359}}$	**0.339**	0.311
TRisk(5)	-6.4	-7.0	-8.6	$\underline{\mathbf{-6.1}}$	-8.8	-7.4	**-7.2**	-7.9	-9.0	**-9.4**	-10.7	**-8.8**	**-10.6**	$\underline{\mathbf{-8.5}}$	**-9.0**	-9.9
M-TRisk(5)	5.6	4.2	-1.3	$\underline{\mathbf{5.3}}$	-1.3	5.1	**5.2**	0.1	6.7	**5.5**	0.7	$\underline{\mathbf{7.3}}$	**5.2**	**6.1**	**4.4**	4.6
B-TRisk(5)	1.3	-0.13	-2.7	$\underline{\mathbf{1.3}}$	-2.7	-0.1	-0.2	-1.8	0.1	-0.6	-2.3	$\underline{\mathbf{0.3}}$	-1.1	-0.1	-0.7	-0.9
FRisk↓	0.163	0.197	0.299	$\underline{\mathbf{0.163}}$	0.303	0.194	**0.178**	0.26	0.179	**0.191**	0.239	**0.176**	**0.205**	$\underline{\mathbf{0.172}}$	0.196	0.213
Loss>20%↓	28	32	39	$\underline{27}$	40	33	30	39	47	50	55	$\underline{46}$	51	$\underline{46}$	49	50
Win	9	7	1	$\underline{9}$	1	7	$\underline{9}$	3	14	15	5	$\underline{14}$	7	$\underline{14}$	$\underline{14}$	12

Table 2: *The risk-sensitive evaluation measured in TD2003 and TD2004 datasets, using the Random Forest algorithm.*

	Full	$\overset{E}{\succ}$	$\overset{E\text{-}F}{\succ}$	$\overset{E\text{-}R}{\succ}$	$\overset{T\text{-}F}{\succ}$	$\overset{T}{\succ}$	DivFS	BTFS
NDCG@10	0.445	0.441	0.433	$\underline{\mathbf{0.444}}$	0.438	0.441	0.435	0.433
TRisk(5)	-87.5	-89.407	-92.297	$\underline{\mathbf{-88.279}}$	-91.026	-89.986	-92.416	-92.74
FRisk	0.147	0.15	0.155	$\underline{\mathbf{0.148}}$	0.152	0.151	0.154	0.156
Losses>20%↓	4695	4790	4944	$\underline{4742}$	4899	4835	4947	5001
Win	2189	2094	1910	$\underline{2136}$	1979	2052	1911	1920

Table 3: *Risk-sensitive evaluation using LambdaMART (and the MAX baseline) in the WEB10K dataset.*

5. CONCLUSION

In this paper, we proposed a multi-objective FS method capable of optimizing important aspects of the L2R task. We used our method to evaluate different combinations of objectives, including risk-sensitive evaluation which, to the best of our knowledge, has never been considered in feature selection tasks. Our proposed extensions of SPEA2 (regarding a strict comparison among individuals, and the use of a biased L2R method) enabled the creation of a competitive feature selection approach based on a multi-objective criteria. In fact, we were able to successfully guide the search for the best known subsets of features regarding ranking performance and low-risk in all evaluated datasets.

We evaluated our proposals considering different objective criteria (i.e., the number of features, ranking performance, and risk-sensitive tests) and taking into account important state-of-the-art baselines. Our experimental results show that the reduction in the risk in some queries as an explicit objective is essential to produce robust models. However, we also reduced dimensionality significantly with some robust models, without harming the overall effectiveness. In addition, our proposal is capable of finding feature subsets that can maximize the ranking performance and minimize the risk and which are highly consistent over different datasets. In brief, we were able to answer our main research question positively: *is there a subset of features capable of maximizing effectiveness and minimizing risk?* Our experiments show that there is a subset of features with over 17% and 26% reduction, with a ranking performance and the risk-sensitive evaluation similar to the full features.

As future work, we plan to use other general-propose multi-objective methods. We also intend to exploit the proposed multi-objective feature selection approach in different applications, such as recommender systems.

6. ACKNOWLEDGMENTS

This work was partially funded by projects InWeb (grant MCT/CNPq 573871/2008- 6) and MASWeb (grant FAPEMIG/ PRONEX APQ-01400-14), and by the authors' individual grants from CNPq, CAPES, FAPEMIG, FAPEGO, and IFG.

7. REFERENCES

[1] L. Breiman. Random Forests. *Machine Learning*, 2001.

[2] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. *SIGIR*, 1998.

[3] A. M. et al. Web-search ranking with initialized gradient boosted regression trees. *JMLR*, 2011.

[4] B. L. et al. Many-Objective Evolutionary Algorithms: A Survey. *ACMCS*, 2015.

[5] B. P. K. et al. Explaining the user experience of recommender systems. *UMUAI*, 2012.

[6] B. T. D. et al. Hypothesis testing for the risk-sensitive evaluation of retrieval systems. *SIGIR*, 2014.

[7] B. T. D. et al. Tackling Biased Baselines in the Risk-Sensitive Evaluation of Retrieval Systems. *ECIR*, 2014.

[8] D. D. et al. Quality Assessment of Collaborative Content With Minimal Information. *JCDL*, 2014.

[9] E. Z. et al. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. *EMDOCAIP*, 2001.

[10] F. P. et al. Greedy and randomized feature selection for web search ranking. *ICCIT*, 2011.

[11] G. G. et al. Is Learning to Rank Worth it? A Statistical Analysis of Learning to Rank Methods in the LETOR Benchmarks. *JIDM*, 2013.

[12] H.-J. L. et al. FSMRank: feature selection algorithm for learning to rank. *IEEE TNNLS*, 2013.

[13] H. P. et al. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE PAMI*, 2005.

[14] J.-G. H. et al. Preliminary study on Wilcoxon learning machines. *IEEE TNN*, 2008.

[15] K. C.-T. et al. TREC 2013 Web Track Overview. In *TREC 2013 web track guidelines*, 2013.

[16] L. J. W. et al. Pruning and ranking the Pareto optimal set, application for thedynamic multi-objective network design problem. *JAT*, 2012.

[17] L. L. et al. Nonconvex regularizations for feature selection in ranking with sparse SVM. *IEEE TNNLS*, 2014.

[18] L. W. et al. Robust ranking models via risk-sensitive optimization. *SIGIR*, 2012.

[19] M. K. N. et al. *Applied linear statistical models.* 1990.

[20] M. L. et al. On the effects of archiving, elitism, and density based selection in evolutionary multi-objective optimization. *EMCO*, 2001.

[21] P. Z. et al. Generalized Bias-Variance Evaluation of TREC Participated Systems. *CIKM*, 2014.

[22] T. H. et al. *The Elements of Statistical Learning.* 2009.

[23] X. G. et al. Feature Selection for Ranking. *SIGIR*, 2007.

[24] T.-Y. Liu. Learning to Rank for Information Retrieval, 2007.

[25] K. D. Naini and I. S. Altingovde. Exploiting Result Diversification Methods for Feature Selection in Learning to Rank. *ECIR*, 2014.

[26] T. Sakai. Statistical reform in information retrieval? *SIGIR*, 2014.

[27] M. B. Shirzad and M. R. Keyvanpour. A feature selection method based on minimum redundancy maximum relevance for learning to rank. *AIR*, 2015.

[28] M. Srinivas and L. M. Patnaik. Genetic algorithms: a survey. *IEE CS*, 1994.